Qualification Accredited

OCR
Oxford Cambridge and RSA

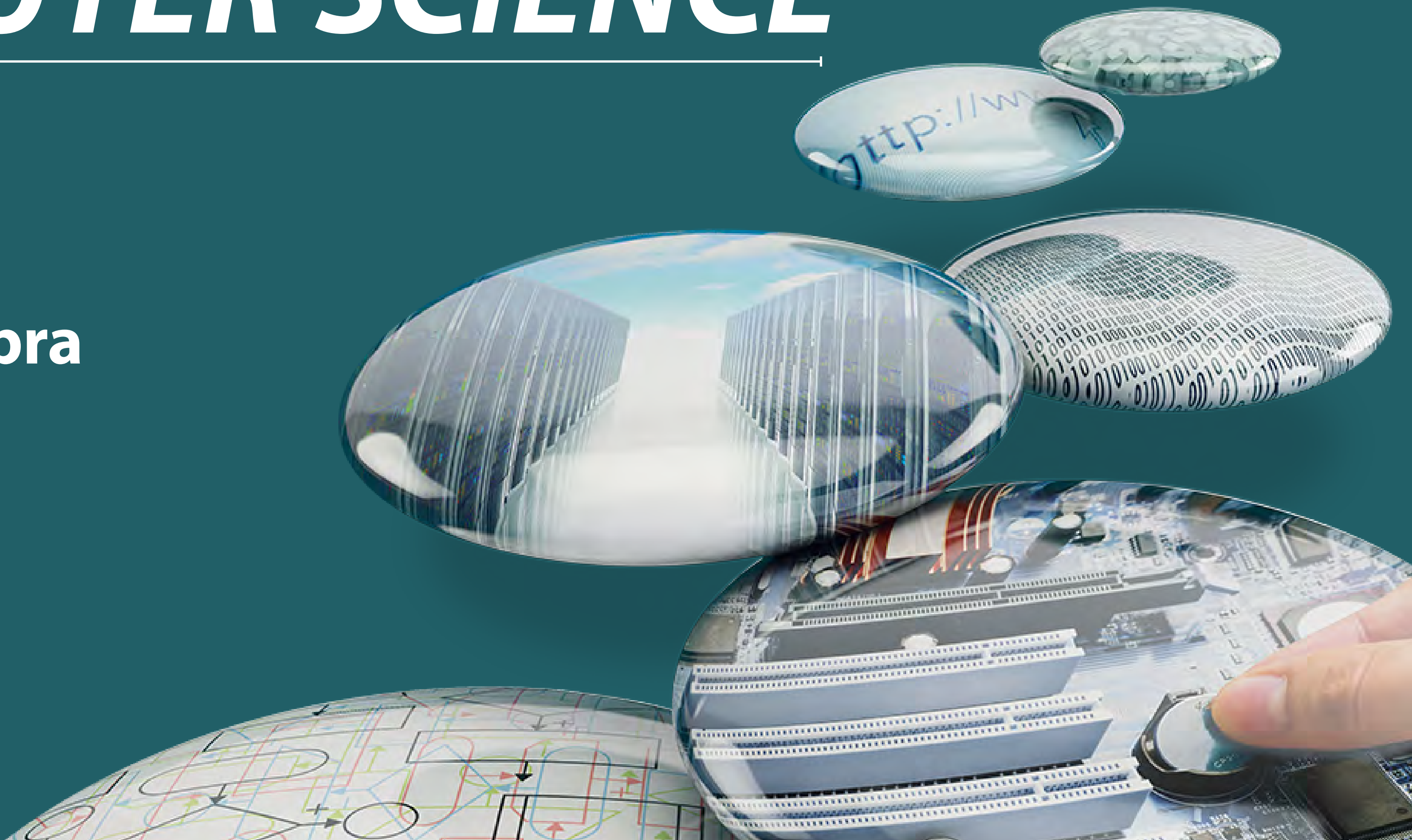
A LEVEL
Delivery Guide

COMPUTER SCIENCE

H446
For first teaching in 2015

Theme: 1.4.3
Boolean Algebra

Version 2



www.ocr.org.uk/computerscience

CONTENTS

Introduction

Delivery guides are designed to represent a body of knowledge about teaching a particular topic and contain:

- Content: a clear outline of the content covered by the delivery guide;
- Thinking Conceptually: expert guidance on the key concepts involved, common difficulties students may have, approaches to teaching that can help students understand these concepts and how this topic links conceptually to other areas of the subject;
- Thinking Contextually: a range of suggested teaching activities using a variety of themes so that different activities can be selected that best suit particular classes, learning styles or teaching approaches.

If you have any feedback on this Delivery Guide or suggestions for other resources you would like OCR to develop, please email resources.feedback@ocr.org.uk.

KEY

Click to view associated resources within this document.

Click here — Click to view external resources

AS Level only — AS Level content only

Curriculum Content

a) Define problems using Boolean logic. See appendix 5e of the specification.

b) Manipulate Boolean expressions. Including the use of Karnaugh maps to simplify Boolean expressions.

c) Use the following rules to derive or simplify statements in Boolean algebra: De Morgan's Laws, distribution, association, commutation, double negation.

d) Using logic gate diagrams and truth tables. See appendix 5e of the specification.

e) The logic associated with D type flip flops, half and full adders.

Thinking Conceptually

Boolean algebra is a sub-area of mathematical algebra where the only values that can be represented are TRUE or FALSE (1 or 0). Knowledge of Boolean algebra is fundamental in programming and the design of modern digital equipment, and it is also used in set theory and statistics. Electronic engineering courses will teach Boolean algebra as standard.

A good approach for teaching Boolean Logic would be to start with the idea of logic gates (Content section d) and truth tables, as these are concepts that can be easily covered and can therefore be used to build students' confidence with the new material. You may choose to combine practice examples and questions with constructing the logic circuits on a simulator such as 'Logic gate simulator' by Steve Kollmansberger (http://www.kolls.net/gatesim/).

Click here

At this stage, you can then introduce the idea of defining problems with Boolean Logic (Content section a). You can relate these ideas to real life (e.g. having to take Physics and either Geography or History as subject options) or in terms of machines (e.g. the CTRL, ALT and DEL keys should be pressed all at once). You could then start to look at how different logic circuits can have the same output, and how we might choose to define whole logic circuits using only NAND gates to save money, since NAND gates can be used to emulate any other gate.

You could then look at Content section b, dealing with manipulating Boolean expressions, firstly teaching simple Boolean simplification laws such as basic Boolean identities and drawing truth tables for these. All of this can be found in Learner Resource 1 (on page 10).

Learner Resource 1

After this you might cover Content section c– the laws of commutation, association, distribution and double negation. These are much the same as laws in Mathematics such as A+B is the same as B+A and (A+B)+C is the same as A+(B+C). A good idea would be to then teach the laws of DeMorgan's Law for algebra simplification. A rundown of these can be found in Learner Resource 1 (on page 10).

Learner Resource 1

You then might choose to cover Karnaugh maps as something students should understand, but mostly students might prefer to use truth tables as a way to check whether they have manipulated/simplified their expressions correctly. They may use truth tables as a way of simplification itself, as they may be able to draw conclusions by looking at the output.

Flip-flops, and half and full adders are a bit different from usual logic gate examples and are best taught at the end once students have acquired a full grasp of Boolean logic.

Common misconceptions or difficulties students may have

Boolean algebra is quite a complex topic for those who have not encountered it before and will therefore display quickly the range of abilities in the class.

If students struggle with mathematics they may have trouble getting their heads around distributing Boolean expressions or factoring them out. To help students understand these, you could ask them to produce the circuits in simulators or to complete truth tables to prove each law. Students may find that checking their own simplifications with truth tables before being shown the answer gives them the will to find the answer for themselves, as they will know that they are wrong and can revisit their workings to find out why.

In particular, students sometimes find the second distributive rule A+(B.C) = (A+B).(A+C) quite difficult to master as it does not follow

Learner Resource 1

the same conventions as normal mathematics, where this would not usually factor out. This is why it is important to cover where the rule comes from. The workings for this can be found in Learner Resource 1 (on page 10).

Students can have trouble recognising where certain rules can be applied. This can sometimes be due to something as simple as the letters being used, such as XY as opposed to AB. In this case, if students really struggle they might want to replace the letters with ones they are more comfortable with, so long as they change them back afterwards!

Students' learning and understanding will benefit through having worked-through examples to reference and lots of practice. Parallels could be drawn with puzzles such as Sudoku, in that they require you to have lots of practice to become versed in doing the calculations and complete them with speed.

It would be a good idea to weave in and out of this topic with something that is a bit less taxing. It might, for example, be a good idea to cover a topic like 1.5.2 Ethical, Moral and Cultural Issues at the same time so that this topic has a chance to sink in and students' interest in it is maintained.

This tactic may also give teachers some time to get exemplar questions completed for homework or as lesson starters/plenaries to help students remember and, if any students have large problems with the topic from a mathematical perspective, to provide them with additional help.

Conceptual links to other areas of the specification – useful ways to approach this topic to set students up for topics later in the course

In concept, specification area 1.4.1 Data Types relates to Boolean algebra as both are to do with how computers deal with logic and numbers. However, for simplicity and teaching, both can be treated as separate entities. You would not want to teach both parts at the same time as there would be too much complexity for students to handle all at once.

Activity	Resources
Interpreting and constructing logic diagrams Learner Resources 1 and 2 can be used together to introduce how you might go between Boolean algebra equations and a circuit diagram. Learner Resource 1 starts off by giving a few examples and then asks some questions that the learners should have a go at answering. Learner Resource 2 can be used for learners to cut out and create their own diagrams so that they can have a go at drawing the circuit into their books, creating a Boolean equation from the diagram and then constructing a truth table. They may then go on to fully prove whether their logic is correct by using a program such as Logic gate simulator (see 'Thinking conceptually' above).	Learner Resource 1 Learner Resource 2
Karnaugh maps Learner Resource 3 is an introduction to Karnaugh maps with some worked examples. You could use this to introduce the topic. After studying this worksheet you may choose to set some further problems for students. There are some worked examples at: http://www.facstaff.bucknell.edu/mastascu/elessonshtml/logic/logic3.html A helpful website with examples of how to (and how not to!) properly group cells can be found at: http://www.ee.surrey.ac.uk/Projects/Labview/minimisation/karrules.html	Learner Resource 3 Click here Click here

Thinking Contextually

Unless the student has previously studied computer science, it is unlikely that they will have come across logic gates or Boolean logic before, and we are unlikely to encounter them outside the realm of the subject area.

It is suggested that lots of examples are given and, where possible, examples are related to real life as opposed to just 0s and 1s, such as "To launch a missile, the operator must turn the key (logic 1) AND press the button (logic 1)".

Learner Resource 1 Interpreting and constructing logic diagrams

See page 8

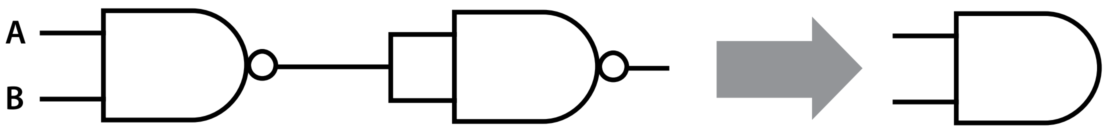
The reason we want to simplify logic is to be able to use less hardware, or, if possible, the same piece of hardware repeatedly. NOR and NAND gates are quite commonly used to create logic simplifications, since they can be configured to act like any logic gate. You can usually buy 4 NAND or NOR gates in one chip, for example. At the time of writing, a QUAD NAND on a quick look was about 30 pence. If you really want to, you could try and create some of the circuits you study with actual components.

We will have a quick look at how this is possible now.

NOT equivalency

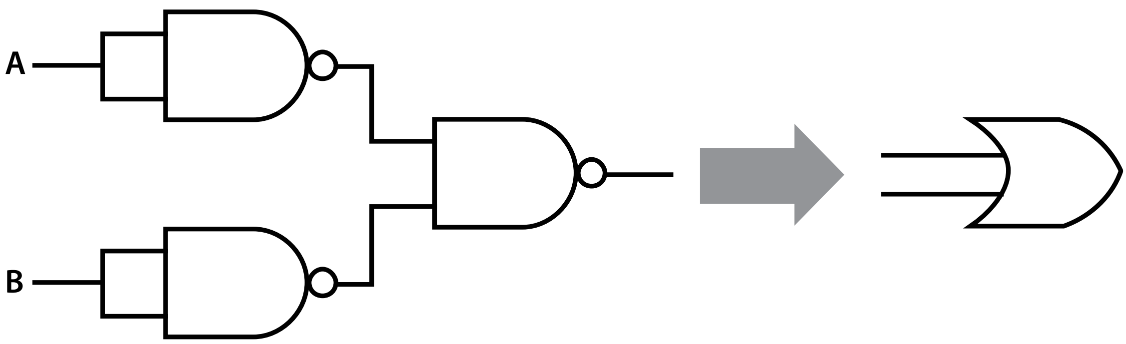
The logic is $\overline{A.A}$, which, as we know from simplification, is $\bar{A}$

AND equivalency



The logic is $\overline{\overline{A.B}}$, so, through double negation, we are left with A.B

OR equivalency



The logic of this is:

$\overline{\bar{A}.\bar{B}}$ using De Morgan's theorem, $\bar{\bar{A}} + \bar{\bar{B}}$, and using double negation we are left with A + B

Going from logic equation to diagram

1) Draw a logic diagram for $Q = \bar{A}.\bar{B} + C$. Prove it using a logic simulator and truth table.

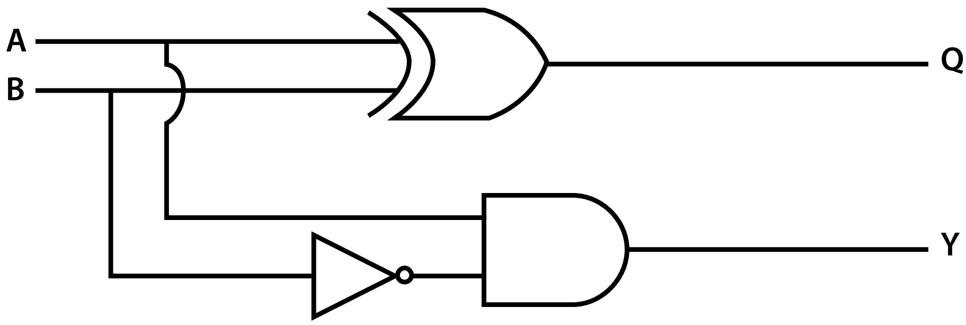
2) Draw a logic diagram for (A+B).(A+C). Prove it using a logic simulator and truth table.

3) Draw a logic diagram for A + (B.C). Prove it using a logic simulator and truth table.

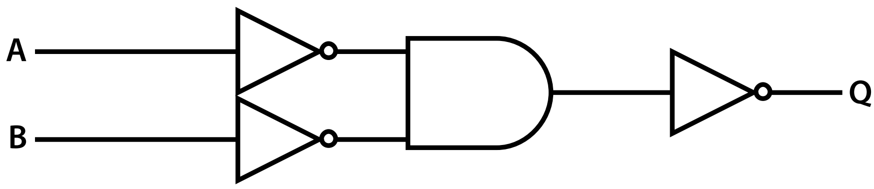
4) Which Boolean simplification law is represented by questions 2 and 3?

Going from diagram to Boolean equation

1) Write down the equations for Q and Y for the diagram below.

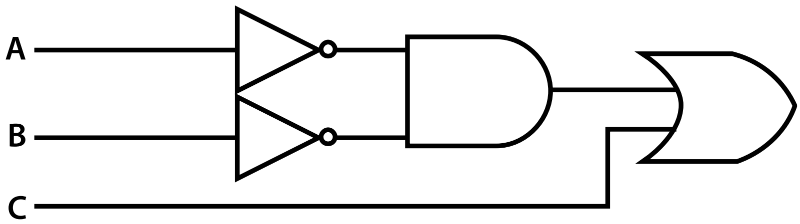


2) Write down the equations for Q for the diagram below.

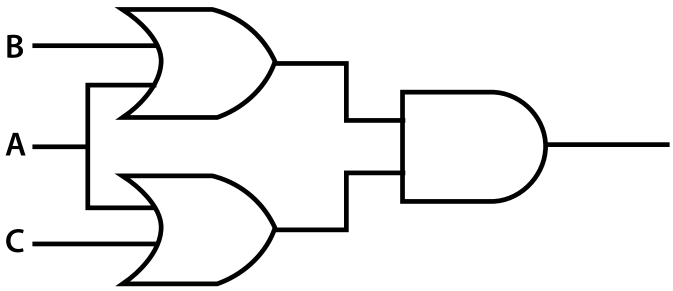


Going from logic equation to diagram

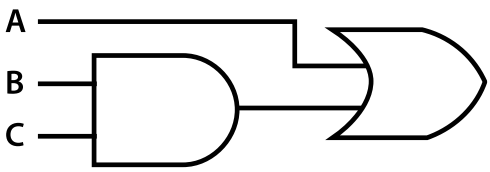
1) Draw a logic diagram for Q = $\bar{A}.\bar{B}$ + C Prove it using a logic simulator and truth table.



2) Draw a logic diagram for (A+B).(A+C). Prove it using a logic simulator and truth table.



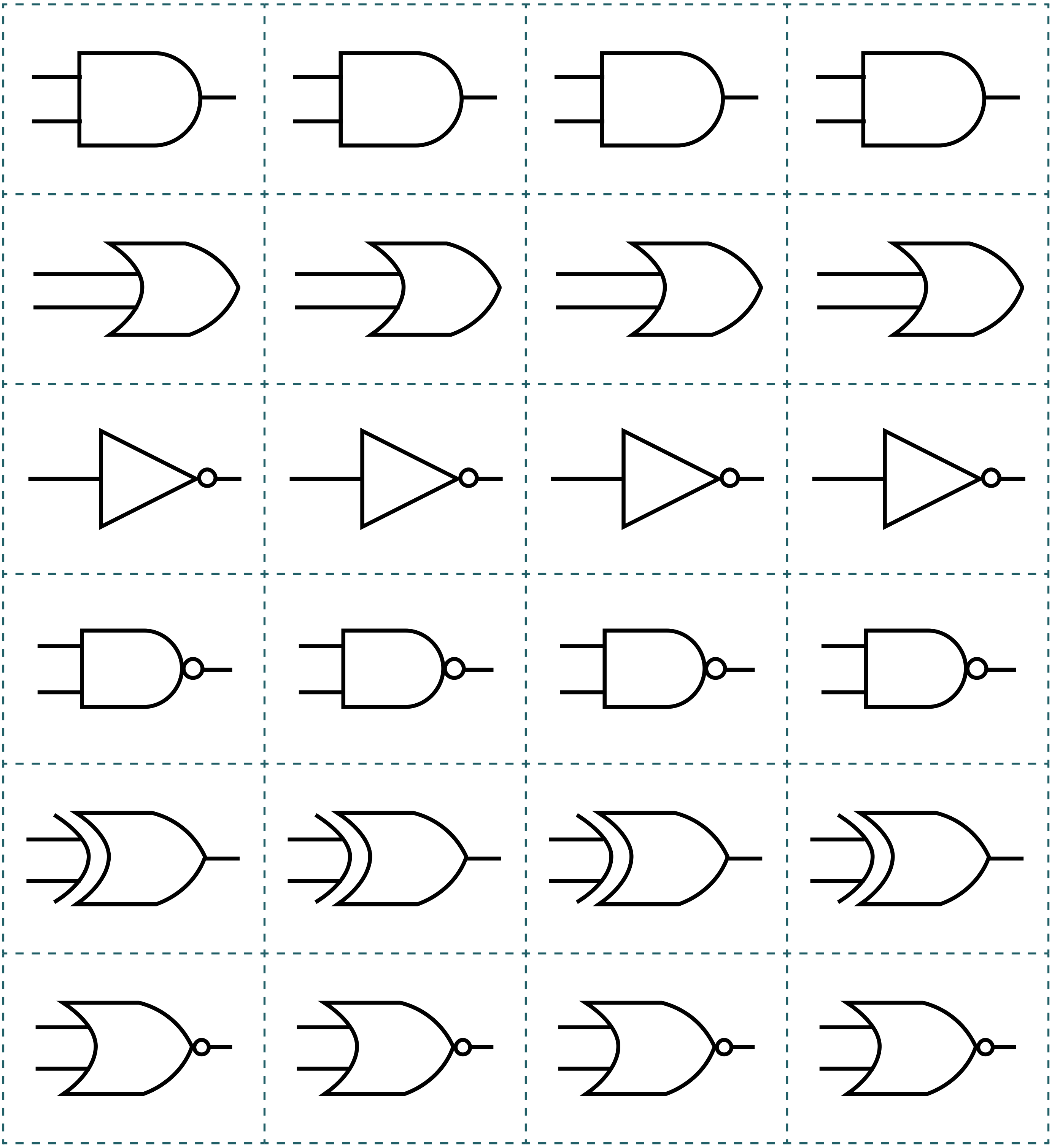
3) Draw a logic diagram for A + (B.C). Prove it using a logic simulator and truth table.



4) Which Boolean simplification law is represented by questions 2 and 3?

Learner Resource 2 Logic card symbols

See
page 8



Learner Resource 3 Karnaugh maps

See page 8

Karnaugh maps (sometimes called K-maps) are used as a way to simplify Boolean algebra expressions. Truth tables and manipulating Boolean expressions using rules are other methods we have available, but what makes Karnaugh maps different is the ability for us to quickly recognise patterns.

Consider the truth table:

A	B	C	D	Q (Output)
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

AB \ CD	00	01	11	11
00	0	0	0	0
01	0	0	0	0
11	1	1	1	1
10	0	0	0	0

Because we can see that there is a block of 1s together, then we see that the 1s on the output only happen if A and B are 1 according to the Karnaugh map. **Therefore we can say that Q = AB**.

So, why aren't Karnaugh maps in Boolean order?

Instead of the digits being in Boolean order, the digits are in order of **Gray Code**. Gray code is a system whereby two successive values differ by only 1 bit.

Look at this map on the left in Gray code and compare it to the one on the right that is in binary code.

AB \ CD	00	01	11	10
00	0	0	0	0
01	0	1	1	0
11	0	1	1	0
10	0	0	0	0

AB \ CD	00	01	10	11
00	0	0	0	0
01	0	1	0	1
10	0	0	0	0
11	0	1	0	1

In the first diagram we can tell straight away that Q = BD, but in the second diagram it is harder to tell because the transition from 01 to 10 changes two bits, and this does not help to group similar values.

Rules for K-maps

- Groupings must be square/rectangular.
- We want to make groupings as large as possible. We can overlap them in order to make the groups bigger.
- They have groups of powers of 2 (1, 2, 4, 8 etc.)

Further Examples

Example 1

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	0	0	0	1
10	0	1	1	1

For the red square, D is 1 whilst A is 0 so we have $D\bar{A}$.

For the green rectangle we know that this will be $C\bar{D}$.

For the blue rectangle this will be $AD\bar{B}$.

Therefore the simplification is $Q = D\bar{A} + C\bar{D} + AD\bar{B}$.

Example 2

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	1	1
11	1	1	1	1
10	1	1	1	1

The red rectangle is C.

The blue rectangle is A.

Therefore Q = A+D.

Example 3: 3 inputs

AB \ C	0	1
00	0	0
01	0	1
11	0	1
10	0	1

The temptation would obviously be to group together the three 1s, but our groupings must be a power of 2.

We therefore should use two overlapping groups of 2, as this is the largest that we can make them.

The green group is BC.

The blue group is AC.

Therefore the output Q = BC + AC.

The small print

We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

Whether you already offer OCR qualifications, are new to OCR, or are considering switching from your current provider/awarding organisation, you can request more information by completing the Expression of Interest form which can be found here: www.ocr.org.uk/expression-of-interest

OCR acknowledges the use of the following content:
Square down and Square up: alexwhite/Shutterstock.com

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: resources.feedback@ocr.org.uk

Looking for a resource?

There is now a quick and easy search tool to help find **free** resources for your qualification:

www.ocr.org.uk/i-want-to/find-resources/

ocr.org.uk/alevelreform
OCR customer contact centre

General qualifications

Telephone 01223 553998
Facsimile 01223 552627
Email general.qualifications@ocr.org.uk

OCR is part of Cambridge Assessment, a department of the University of Cambridge. *For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored.*